Formation Unix/Linux - administration

Guillaume Allègre
Guillaume.Allegre@silecs.info

SILECS
INPG Formation Continue

INPG FC

mai 2008

Installation d'un système Linux

Préparatifs

Avant l'installation

- Choix d'une distribution Linux... selon contraintes
- Espace disque pour l'installation du système
  - Pas de problème si Linux est seul
  - A répartir si plusieurs systèmes coexistent
- Vérifier que le matériel est compatible
  - carte vidéo, carte mère, carte réseau
  - Noter les caractéristiques des périphériques peu communs
- Configuration du réseau : environnement

Dualboot Windows-Linux

- Windows + Linux est plus facile que Linux + Windows
- Dégager de l'espace libre (*Partition Magic*, *parted*...)
- Prévoir des partitions communes aux deux systèmes ?

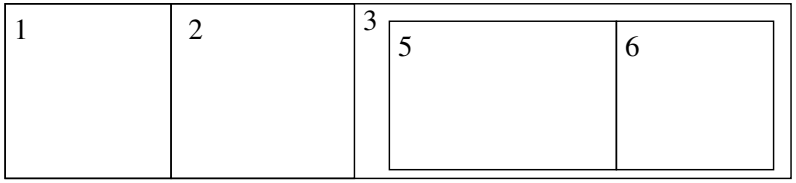
Nommage disques durs et partitions

"Numérotation" des disques durs

- Disques IDE (PATA/SATA) : `/dev/hda`, `/dev/hdb`...
- Disques SCSI (// ou SAS) : `/dev/sda`, `/dev/sdb`...

Numérotation des partitions

- Partitions 1 à 4 : primaires (dont au max. une étendue)
- Partitions 5+ : logiques
- Exemple :



Systèmes de fichiers

Les différents types de systèmes de fichiers

- Systèmes Linux
  - ext2fs : système unix simple, rapide
  - ext3fs : ext2fs + journalisation (fiabilité)
- Systèmes externes
  - FAT32, NTFS : systèmes Windows (double-boot)
  - ReiserFS : autre système Unix libre, "évolué"
  - XFS : système d'origine IBM

Mais aussi...

- SWAP : partition d'échange mémoire (1x à 2x RAM)
- Implémentation : modules noyau Linux

Choix d'une distribution : Debian

Debian

- distribution communautaire non commerciale
- très grand nombre de paquets officiels
- excellente gestion des dépendances et des mises à jour
- toujours 3 "suites" Debian en cours
  - stable (act. 4.0r3 *Etch*, sortie le 8/04/2007, r.3 le 17/02/08)
  - testing (act. *Lenny*)
  - unstable (*Sid*)
- réputation d'être moins *user-friendly* que la concurrence

L'installateur est assez rudimentaire et demande des compétences techniques.

Installation Debian

Plusieurs supports possibles

- CD-ROM / DVD-ROM
- CD-ROM puis réseau
- Clé USB puis réseau
- Tout-réseau : serveur TFTP (PXE / Etherboot)

Installations automatisées

- Préconfiguration (*preseed*)
- FAI (*Fully Automated Installation*) : serveur nécessaire
- Cf Manuel d'Installation, 4.7

Les options de démarrage

Plusieurs méthodes

- install (par défaut)
- expert (plus de questions)
- rescue (sauvetage / restauration)
- pour chacune : mode texte ou GUI

Multiples paramètres d'amorçage

- paramètres pour l'installateur Debian
- gestion du matériel problématique : paramètres des modules Linux

Exemple

- `expert-gui locale=fr_FR acpi.blacklist=yes`

Pendant l'installation

Phase 1 : installation

- Accès aux consoles : shell, log installateur, log noyau

Phase 2 : redémarrage et configuration

- debconf : base de données des configs de paquets

Cohabitation avec d'autres systèmes

boot manager (GRUB/LILO) : choisir le système d'exploitation au démarrage.

Partitions

- Linux peut accéder aux partitions Windows
  L'accès en écriture est par défaut restreint avec NTFS.
- Windows ne peut pas accéder aux systèmes de fichiers Linux.
  Des extensions le permettent (non recommandé).

Ne pas confondre avec Samba (serveur de fichiers).

Bases de l'administration système

Comptes utilisateurs

Fichiers concernés

- `/etc/passwd` et `/etc/shadow`
- `/etc/group` et `/etc/gshadow`
- `/etc/skel/`
- `/etc/shells`

Commandes

- useradd / userdel (standard, paquet `passwd`)
- adduser / deluser (extension Debian) + `/etc/adduser.conf`
- passwd

Création des comptes :

- manuelle : modification `/etc/passwd`, `/etc/shadow`...
- `adduser john` interactive
- `adduser john ...` en ligne de commande

Anatomie des fichiers de configuration

/etc/passwd

1. nom de connexion de l'utilisateur (login)
2. mot de passe chiffré (ou x $\Longrightarrow$ cf `shadow`)
3. identifiant numérique de l'utilisateur (UID)
4. identifiant numérique du groupe principal de l'utilisateur (GID)
5. nom complet + commentaires (Gecos)
6. répertoire personnel de l'utilisateur
7. shell de l'utilisateur (ou `/usr/sbin/nologin`)

/etc/shadow

1. login utilisateur
2. mot de passe chiffré MD5
3. 7 paramètres de validité du mot de passe (`man 5 shadow`)

La commande `chage` : modifier les paramètres de validité

Les sudoers

- Le fichier de configuration : `/etc/sudoers`
  - des définitions d'alias
  - des autorisations : UTILISATEUR HOTE = (EN-TANT-QUE) COMMANDE
- Les commandes utilisateurs :
  - `sudo <commande>`
  - `sudoedit <fichier>` ou `sudo -e <fichier>`

Exo

1. Accorder à l'utilisateur par défaut les droits de root
2. Autoriser un groupe "grh" à créer et supprimer des comptes.

PAM : Pluggable Authentication Modules

- Principe
  - une infrastructure d'authentification unifiée
  - partagée entre le système et les applications
  - un jeu de modules d'authentification
  - extensible et paramétrable par l'administrateur
  - commun à plusieurs Unix : Linux, FreeBSD, Sun (origine), HP-UX
- Paquet de base Debian : `libpam-modules`
- Documentation
  - The PAM FAQ
  - The Linux-PAM System Administrators' Guide, v1.0
  - The Linux-PAM Module Writers' Guide
  - The Linux-PAM Application Developers' Guide

Séquence de démarrage (boot)

Après la mise sous tension, le BIOS exécute le programme placé au début du périphérique de boot.

Gestionnaires de démarrage standard

- GRUB (GRand Unified Bootloader
- LILO (LInux LOader) : plus simple

Un *boot manager* permet

- de choisir le système d'exploitation (boot sur une partition donnée)
- de charger un noyau Linux avec les paramètres voulus (plusieurs noyaux peuvent cohabiter)

Démarrage Linux

1. Gestionnaire de boot (GRUB/LILO)
2. Chargement du noyau
   La console texte permet de suivre l'évolution (diagnostic)
3. init est le premier processus lancé
4. init exécute les scripts d'initialisation de `/etc/init.d`

init

init : premier processus
Appelé par le noyau (avec en argument optionnel un run-level / initlevel)

Runlevels

- 0 extinction
- 1 *single user* (dépannage, root seulement)
- 2-5 niveaux utilisateurs
- 6 redémarrage
- S boot (unique)

Les niveaux 2 à 5 sont personnalisables par l'administrateur.

Configuration : `/etc/inittab`
Répertoires associés : `/etc/rc?.d`

init : un exemple

Scénario de démarrage sans paramètre noyau

- Linux lance init
- Le run-level n'est pas fixé, donc *initdefault* de `/etc/inittab`
  $\Longrightarrow$ run-level=2 (Debian...) ou 5 (RedHat...)
- init lance les consoles textes
- Pour chaque lien de type `/etc/rc5.d/K??script`, init arrête le service en lançant `script stop`.
- Pour chaque lien de type `/etc/rc5.d/S??script`, init démarre le service en lançant `script start`.

Anatomie d'un script init.d/...

TP – Manipulation des runlevel

1. Vérifier le run-level actuel (`runlevel`)
2. Passer en run-level 2.
3. Lancer le mode graphique manuellement.
4. Tuer le *getty* d'une console. Que constate-t-on ?
5. Repasser en mode de départ. Conclusion ?

Les services : cron

cron : lancement automatisé de tâches

- cron démon (*daemon*) : programme résident en mémoire qui réalise les tâches de fond du système.
- Les tables de tâches (crontab) utilisateurs
- Les tables système ...

Cron utilisateur

- fichier de configuration : `crontab -e`
- Syntaxe : m h dom mon dow command
  (cf `man 5 crontab`)

Exo

1. Ajouter la date dans le fichier `timestamp` toutes les cinq minutes.

Les crontab système

- `/etc/crontab` (+ champ User)
- `/etc/cron.daily`, `etc/cron.weekly`...
- `/etc/cron.d/*`

Exemples

- `/etc/cron.daily/find` et `locate`
- `/etc/cron.daily/dlocate` et `dlocate`
- Démon `anacron` : gère les interruptions

Les logs

Tous les événements importants sont consignés dans `/var/log/`.

- soit via syslog
- soit directement par les applications

les services (démons) : klogd et sysklogd

- `klogd` : reçoit les messages du noyau et les passe à sysklogd
- `syslogd` : reçoit les messages et les diffuse...

Consultation des logs

- dmesg *(noyau : boot + modules)*
- last, lastlog *(connexions utilisateurs)*
- tail (-f), less, grep...

Le service syslog

Composition d'un message

- priorité : 0=debug ... 3=warning ... 5=crit ... 7=emerg
- service (*facility*) (auth mail kern local[0-7] ...)
- texte

Le fichier `/etc/syslog.conf`

- sélecteur : <service>.<priorité>
- action : envoi vers
  - fichier, ex. `/var/log/messages`
  - terminal (ou pseudo-term), ex. `/dev/tty8`
  - machine distante (syslog), ex. `@loghost.localdomain`
  - utilisateurs, ex. `root,john` ou tout le monde, `*`
- `logger` : test *(envoie un log)*

Exo

1. Afficher les logs d'authentification sur la console 8.
2. Horodatage de `/var/log/syslog` toutes les 5 minutes.

Outils complémentaires

Outils automatiques

- logrotate
- logcheck (par défaut sous Debian)
- logwatch (par défaut sous RedHat)

Aide à l'analyse des logs

- LIRE (logreport)
- ...

syslog-ng peut remplacer klogd + syslogd.

TP - Gestion des systèmes de fichiers 1

Exo 1 : résumé du système de fichiers

1. Trouver le nb d'entrées de répertoire de chaque type sous /, sans changer de système de fichiers (`-xdev`).
2. Transformer en script prenant en argument le système de fichiers de départ
3. Pour les quatre types minoritaires, afficher les entrées

Exo 2 : un nouveau montage

1. créer une partition de quelques Go en Ext2fs (avec fdisk...)
2. la rattacher au système de fichiers sur `/mnt/vol`
3. pérenniser ce montage : optionnel, activé par l'utilisateur
4. passer la partition en Ext3fs
5. définir le montage par son label de partition

Commandes : `find, fdisk, mkfs, mount, tune2fs, e2label`
Fichiers : `/etc/fstab`.

TP - Gestion des systèmes de fichiers 2

Exo 3 : un nouvel espace de SWAP

1. créer une nouvelle partition de SWAP (avec parted)
2. l'activer
3. pérenniser cette configuration

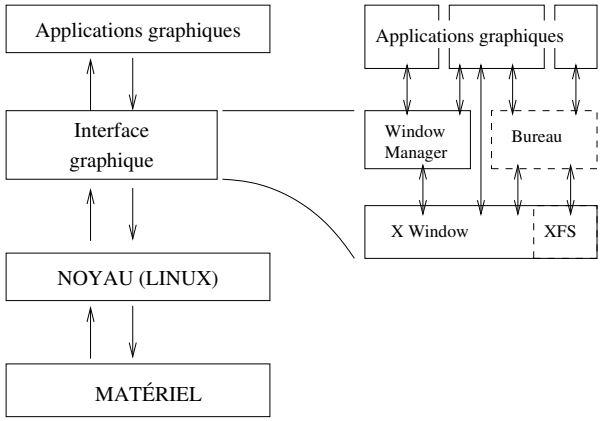
Exo 4 : utiliser des CDs sans lecteur de CD

1. récupérer l'image ISO d'un CDROM (physique)
2. monter localement l'image dans `/media/image`
3. pérenniser cette configuration

Exo 5 : utilisation de Partimage

1. copier quelques répertoires sur la nouvelle partition (exo 2)
2. sauvegarder son image avec `partimage`
3. vandaliser le contenu puis restaurer l'image

Le système de fenêtrage X-Window (X11)



- Système standard sur tous les Unix (sauf NextStep, Mac OS X)
- Transparence réseau (presque) totale
- Architecture client-serveur !

X-Window : historique

Historique

- juin 1984 : X1, MIT
- jan. 1985 : X6, première version diffusée (propriétaire)
- sep. 1985 : X9, couleur, licence MIT
- sep. 1987 : X11, protocole courant
- mai 1994 : X11R6
- déc. 2005 : X11R6.9 + X11R7
- sep. 2007 : X11R7.3

Implémentations libres

- XFree86 : 1992 - 2003 (dissolution de l'équipe) - 2008 ...
- X.org : fork en 2004 (XFree86 4.4rc2), plus dynamique

Installation et configuration Debian

Installation

- un métapaquet : `xorg`
- nombreuses dépendances :
  - xserver-xorg...
  - xfonts-...
  - xbase-clients, xutils, xterm...

Configuration

- autoconfiguration à l'installation (détection du matériel)
- `dpkg-reconfigure xserver-xorg`
- `/etc/X11/xorg.conf` : fichier principal (consulter)
- `/etc/X11/*` : fichiers annexes
- `/var/log/Xorg.?.log` : fichier de log

X11 : principales composantes

- Serveur X (`/usr/bin/X`) (+ serveur de fontes X Font Server)
- Gestionnaire de session X (X Display Manager)
  ex. xdm, kdm, gdm, slim...
- Bureau graphique (optionnel)
  ex. Gnome, KDE, XFCE...
- Gestionnaire de fenêtres (Window manager)
  ex. metacity, kwm, xfwm4, twm, awesome...
- Console / émulateur de terminal
  ex. xterm, mlterm, xfce4-terminal...

Concepts et commandes X11

- Évènements X11 (clavier, souris, logiciel)
  `xev` : tester les entrées
- Propriétés et informations
  - Commande `xwininfo`
  - Commande `xprop`
- Ressources X
  - Commande `xrdb (-query -all)`
  - Fichiers `/.Xdefaults` et `/etc/X11/Xresources/*`
- Contrôle des fenêtres
  Commande `xkill`

Sauvegarde et archivage

Rappel : archives

tar (archivage) + gzip / bzip2 (compression)

Sauvegardes

- Historiquement, sur bandes $\Longrightarrow$ accès longs, séquentiels
- images (disque ou partition) : `dd`, `partimage`, `Clonezilla`
- `dump` + `restore` : outil Unix historique de sauvegarde, orienté bandes
- `cpio` : alternative à tar

Applications complètes

- Amanda : disques + bandes, ligne de commande
- BackupPC : disques seulement, interface web
- ...

Compléments : suivi de version et réplication

Suivi de version

Pour les fichiers sensibles, par exemple `/etc/`

- Principe : stocker l'historique des versions successives
- Outils : CVS, Subversion, FSVS...

Réplication

Pour la sécurité et l'intégrité des données, la redondance

- les fichiers de log (via syslog, syslog-ng...)
- les bases de données (serveurs maître et esclaves)
- les annuaires (LDAP...)

TP - Sauvegarde et archivage

dump + restore

- sauvegarde totale de `/etc` avec `dump`
- restauration interactive de `fstab` et `modprobe.d` dans `/mnt/vol/etc`
- sauvegarde d'un système de fichier au niveau 0 (complète)
- modification de quelques fichiers
- sauvegarde incrémentale des différences
- restauration complète

L'impression sous Unix

Matériel : 3 types de connexions

- imprimantes locales (// ou USB)
- imprimantes réseau (interface ethernet)
- imprimantes locales sur un serveur d'impression (réseau)

Services et protocoles

- applicatif : prépondérance de **PostScript** puis **PDF** (Adobe)
- **lpd/lpr** : historique, RFC1179, 1990
  - **lpd BSD** : implémentation historique
  - **LPRng** : (New Generation) réécriture du précédent
- **CUPS** : Common Unix Printing System
  - RFC 2565-2569, 1999 (Novell - Xerox)
  - Easy Software Products (1997-2007), puis Apple
  - protocole IPP, surcouche à HTTP
  - adopté par Mac OS X (exception à la GPL)

Configuration de l'impression

LPD / LPRng

- un démon : `lpd` (TCP port 515)
- un fichier de configuration : `/etc/printcap`
- des commandes : (BSD) `lpr, lpq, lprm, lpc`

CUPS

- un démon : `cupsd` (TCP ports 515 et 631)
- interface web : `http://localhost:631`
- un répertoire de configuration : `/etc/cups/*`
- paquets Debian : `cupsys, cupsys-bsd...`
- surcouches graphiques :
  - GNOME : gnome-cups-manager
  - KDE : kdeprint (uniformise l'accès aux 3 systèmes)

Localisation et francisation

Paramètres régionaux

- choix du clavier
- langue des messages système et des applications
- jeu de caractères
- convention d'affichage (date, monnaie, tri alphabétique...)
- fuseau horaire
- (éventuellement) polices de caractères

Définitions

- **I18N** : (internationalisation) une application est prête à être "traduite"
- **L10N** : (localisation) la traduction est faite pour une langue ou un pays précis

Locales

Jeux de caractères et locales pour le français

- iso-latin-1 (ou iso-8859-1) : **fr_FR**
- iso-latin-9 (ou iso-8859-15) : **fr_FR@euro**
- UTF-8 : **fr_FR.UTF-8**
- fichiers : `/etc/locale.gen`, `/etc/locale.alias`
- commandes : `locale`, `dpkg-reconfigure locales`

Fuseau horaire

- fichier : `/etc/timezone`
- commandes : `dpkg-reconfigure tzdata`

Gestion des paquets avec APT

D'abord... sans APT !

Installateurs

Par exemple :

- Sun java
- Pilotes Nvidia/ATI

Compiler à partir des sources

Très souvent via le protocole des Autotools (autoconf/automake) :

- ./configure
- make
- make install

Les paquets Debian

Paquet binaire (.deb) ou source (.dsc)

Contenu d'un paquet binaire (.deb)

- Archive des fichiers (data)
- Métadonnées (control/control)
  - Descriptions textuelles : courte et longue
  - Section : classement du paquet dans une hiérarchie debian
  - Version
  - Dépendances, conflits, suggestions, recommandations...
  - debtags : indexation du paquet
    Par exemple : network::service, suite::apache
- Utilitaires (control/...)
  - scripts installation / suppression
  - sommes de contrôle (MD5sum)

Travaux pratiques

Examen du paquet dpkg

- À la main
  Commandes : `ar t`, `tar -x`
- Avec l'outil dédié
  Commande : `dpkg-deb`

Deux cas particuliers

- Méta-paquets
  - Paquet "réel" : le .deb existe
  - Paquet de paquets : via les dépendances
  - Exemple : gnome
- Paquets virtuels
  - Paquet virtuel : le .deb n'existe pas
  - Indique un service générique, fourni par plusieurs paquets
  - Exemple : `mail-transport-agent` ; cf `mailman`

dpkg : gestion locale

dpkg manipule les paquets debian (`.deb`) sans accès réseau.

Principales options de dpkg

- `dpkg -i paquet.deb` ⟶ installe
- `dpkg -r paquet` ⟶ désinstalle
- `dpkg -L paquet` ⟶ liste les fichiers du paquet
- `dpkg -S fichier` ⟶ recherche `fichier` parmi les paquets installés

dpkg est souvent nécessaire pour les opérations fines (conflits importants, diagnostic, etc.)

TP – dpkg

1. Installer `ncdu` à partir des sources. En quoi est-ce pénible ?
2. Télécharger le navigateur Opera (`www.opera.com`) et l'installer grâce à `dpkg`.
3. Avec `dpkg`, lister les fichiers installés par Opera.
4. Quels exécutables sont fournis par le paquet **sysvinit** ?
5. Quels sont les paquets actuellement installés sur votre machine ?
6. De quel paquet provient la commande `ifconfig` ?
7. Reconfigurer le serveur mail local.

La famille apt

- synaptic
- aptitude
- apt-get
  - update
  - install
  - ...
- apt-cache
  - search
  - show
  - policy

Fichiers

`/etc/apt/apt.conf.d/`
`/etc/apt/sources.list`

Documentation

`apt-howto-en`, `apt-howto-fr`

TP – apt

1. Mettre à jour sa distribution.
2. Examiner le fichier `/etc/apt/sources.list` et en comprendre la syntaxe.
   Quelle est l'organisation d'un miroir Debian ?
3. Ajouter aux sources APT les dépôts de la distribution testing.
   Que se passe-t-il en cas de demande de mise à jour ?
4. Créer `/etc/apt/apt.conf` afin de fixer la version (*release*) par défaut à stable.
   Retenter une mise à jour.
5. **apt** garde une copie de sauvegarde des paquets téléchargés.
   Effacer ces fichiers.

La dernière évolution : aptitude

- Historique
  1. dselect
  2. apt-get
  3. aptitude
- Interfaces
  - Ligne de commande (sous-commandes compatibles `apt-get`)
  - Interface semi-graphique (ncurses)
- Les avancées d'aptitude
  - un log des opérations : `/var/log/aptitude`
  - distinction paquets : installés automatiquement / à la demande
  - résolution des dépendances : meilleure, plusieurs alternatives
- Documentation : `aptitude-doc-en`, `aptitude-doc-fr`

Reconfiguration d'un paquet

Debconf

- une mémoire des choix de configuration
- interfaces : dialog, readline, n-i, gnome, kde, (editor, web)
- priorités : low, medium, high, critical
- fichier de configuration : `/etc/apt/apt.conf.d/70debconf`
- base : `/etc/debconf.conf`, `/var/cache/debconf/*`
- manpages : `debconf(7)`, `debconf(1)`, `debconf.conf(5)`

Commandes

- `dpkg-reconfigure <paquet>`
- manpages : `dpkg-reconfigure(8)`, `dpkg-preconfigure(8)`

Le suivi de bugs de Debian

BTS : le Bug Tracking System

- `http://www.debian.org/Bugs/`
- intégration à APT : `apt-listbugs`

Déposer un bug

- le paquet `reportbug`

Autres systèmes d'installation

Attention aux conflits !

- Système fourni par une application
  - mise à jour automatique : ex. Firefox
  - extensions et plug-ins : ex. Firefox !
- Gestionnaire de paquet "upstream"
  - CPAN : Comprehensive Perl Archive Network
  - PEAR / PECL : extensions PHP
  - PyPI : Python Package Index

Créer ses propres paquets

Recompiler un paquet source Debian

- Ajouter les dépôts sources à `/etc/apt/sources.list`
- `apt-get build-dep paquet`
- `apt-get -b source paquet`
- `dpkg -i paquet.deb`

Créer ses propres paquets

En "trichant"

- Méthode automatique
  `checkinstall` *(pas toujours fiable)*
- Conversion
  `alien` *(par exemple de .rpm à .deb)*
  perte des dépendances Debian

À la dure

À partir de sources pas encore dans Debian

- `zcat paquet.diff.gz | patch -p0`
- `debchange` ou `vim debian/changelog`
- `chmod u+c debian/rules`
- `dpkg-buildpackage -rfakeroot -us -uc`

Configuration et administration du réseau

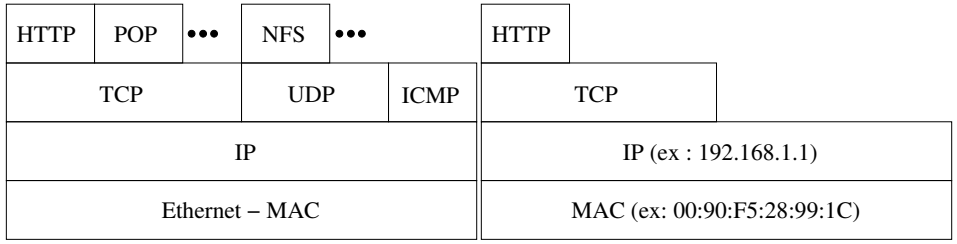
Architecture TCP/IP

Un modèle par couches

Ethernet réseau local Ethernet-MAC

IP l'adressage Internet

TCP le transport



Premières commandes

ifconfig

- lo *(interface virtuelle boucle locale)*
- eth0 *(première interface ethernet)*

- l'adresse MAC : 6 octets
  HWaddr : 00 :90 :F5 :28 :99 :1C
  Propre à la carte réseau
- l'adresse IP : 4 octets, 32 bits
  inet addr : 192.168.1.1

IPv4 & IPv6

IPv6 n'est pas compatible IPv4, donc ici IP=IPv4.

Premiers tests

ping

Tester soi-même, un voisin, un absent, le réseau...

Options utiles

- `ping -c5 192.168.1.1`
- `ping -b 192.168.1.0`
- `ping -f -i 0.2 192.168.1.1`

Exo

1. Changer son adresse IP et retester les pings. Conclusion ?
   `ifconfig eth0 192.168.1.100`
   `ifconfig eth0 192.168.100.1`

Routage, réseau et sous-réseaux

Cheminement d'un message

- Un paquet IP est une partie de message TCP (ou UDP, etc.)
- Dans chaque paquet, 2 adresses IP : source et destination

Anatomie d'une adresse IP

- $\underbrace{192.168.0}_{\text{réseau}}.\underbrace{1}_{\text{hôte}}$ (classe C) ⟵ réseau local
- $\underbrace{172.116}_{\text{réseau}}.\underbrace{0.10}_{\text{hôte}}$ (classe B)

Adresse, masque de réseau, broadcast.

Notation CIDR (Classless Inter Domain Routing)

192.168.0.1/24 ⟶ 24 bits réseau + 8 bits hôte

`ipcalc` : la calculatrice réseaux

Routage : en pratique

Table de routage

Décrit les chemins possibles.

`route -n`

- réseau local
- adresse par défaut (destination 0.0.0.0)

La passerelle (*Gateway*, *Gw*)

Pour sortir du réseau local, la passerelle interconnecte des réseaux.
Souvent X.Y.Z.254

Modifier le routage

`route del default`
Quel impact ?

`route add default gw <ip>` où <ip> est l'ip de la passerelle
Revient à la situation initiale

Les routeurs :
Machines spécialisées avec tables de routage complexes

Suivre une route

`/usr/sbin/traceroute 91.121.14.67`
`mtr (-t|-g) 91.121.14.67`

Config réseau

Config manuelle

```
ifconfig eth0 172.16.0.111
  netmask 255.255.255.0 broadcast 172.16.0.255
route add default gateway 172.16.0.1
```

Config Debian

Dans `/etc/network/interfaces` :

```
iface eth0 inet static
        address 192.168.0.11
        netmask 255.255.255.0
        broadcast 192.168.0.255
        gateway 192.168.0.254

ifdown eth0
ifup eth0
```


La commande `ip`

- la configuration “nouvelle génération” : `ip ss-cmde`
- paquet `iproute`

- ip link : équivalent à ifconfig
- ip address : équivalent à ifconfig
- ip route : équivalent à route

- sous-commandes avancées : multicast, tunnels...

Résolution de noms (DNS)

/etc/hosts

Établit des correspondances *nom d'hôte* ⇆ *adresse IP*

Domaine Name Server (DNS)

- Permet une équivalence entre nom et adresse IP.
  Ex : coriolan.silecs.info ⇆ 82.233.121.16
- Fonctionnement par arborescence de serveurs
  - Dans chaque serveur : cache pour minimiser les requêtes
  - Un *authoritative serveur* fait autorité pour un TLD

Exemples de TLD

- .com .org .edu ...
- .fr .uk .tv ...
- .info .name .museum ...

Fonctionnement du DNS

Modèle client-serveur

Côté serveur

BIND9 très majoritaire

Côté client

Bibliothèque partagée *resolver* dans la *glibc*
Configuration via `/etc/resolv.conf`

- serveurs à interroger (nameserver)
- domaine de recherche par défaut (search)

`/etc/hosts` est prioritaire sur DNS.

Pour affiner les priorités : `/etc/nsswitch.conf`

Clients DNS

- Client léger : `nslookup`
- Clients complets :
  - `dig` (dnsutils)
  - `host` (host)
- Sans oublier...
  `ping` (/etc/hosts puis DNS)

DHCP

Obtenir automatiquement les paramètres réseau

DHCP : client/serveur pour

- adresse IP
- routage (passerelle)
- DNS (facultatif)

Côté client

`dhclient [interface]`
ou `pump -i eth0`

Côté serveur

- Le parc d'adresses est limité
  $\Longrightarrow$ notion de *lease* (bail)
- Contrôle des attributions
  - lier une certaine IP à une adresse MAC
  - autoriser uniquement certaines adresses MAC

Pérenniser la configuration réseau - Debian

Sous Debian
Configuration dans `/etc/network/interfaces`

Au démarrage
Dans `/etc/rc?.d/` lien vers `/etc/init.d/networking`

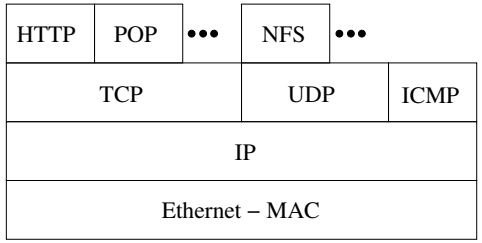
Autres outils
`ifup` et `ifdown` s'appuient sur `/etc/network/interfaces`
Active ou désactive une interface réseau

Récapitulatif

Commandes
- ifconfig
- ping
- route
- traceroute et mtr
- host & dig *(DNS)*
- dhclient & pump *(DHCP)*

Fichiers
- `/etc/hosts` *(DNS)*
- `/etc/resolv.conf` *(DNS)*

Retour sur la pile IP



Passage aux couches supérieures des protocoles (hors ICMP)

Services et ports

Service
Programme côté serveur dans une relation client/serveur
Attaché à un couple port/protocole

Illustration : `/etc/services`
Liste **informative** des services communs

Ports
- désigné par un numéro < 65536
- attaché à un protocole TCP ou UDP
  43/TCP != 43/UDP
- les ports inférieurs à 1024 sont d'accès restreint

TCP / UDP

TCP (Transport Control Protocol)

- orienté *connexion*
  paquets ordonnés, type conversation (*stream*)
- fiabilisé : contrôle & correction d'erreur
- plutôt lent
- le plus utilisé par les services usuels

UDP (User Datagram Protocol)

- paquets indépendants
- plus réactif et rapide
- utilisé par NFS et Netbios (SMB)

inetd : le super-démon

Mode d'exécution d'un service

- démon : lancé indépendamment (`/etc/init.d/...`)
- inetd : lancé à la demande par le super-démon inetd

Exemple : telnet

- Installer telnet et telnetd
- `netstat -a -tu -ep` avec et sans connexion `telnet`
- configuration dans `/etc/inetd.conf`
- Désinstaller telnetd !

Alternative : xinetd

`xinetd` remplace fréquemment `inetd`.

tcpwrapper

Auxiliaire de inetd

Les services `inetd` sont lancés via `tcpwrapper`
Partiellement redondant avec `xinetd`

Son rôle : sécurisation

- Contrôle des autorisations
- Configuration :
  - `/etc/hosts.allow`
  - `/etc/hosts.deny` in.telnetd :ALL

Pour aller plus loin

- `man hosts_access` et `man tcpd`
- Souvent inutilisé, mais cause de bloquages imprévus

Exemple de service : SSH

SSH : connexions sécurisées

Communications cryptées, authentification par clé privée/publique

La famille SSH

- ssh : le client
- sshd : le serveur
  Identification des utilisateurs et des hôtes
- scp : copie de fichiers via ssh
- sftp : émulation ftp via ssh
- ssh-keygen
  Clés utilisateur / hôte

Remarques

- Protocole 1 (obsolète) ou 2
- X11 forwarding : ouverture à distance d'applis graphiques